

This document is published at:

Ruiz-Iniesta, A., Melgar, L., Baldominos, A. y Quintana, D. (2018). Improving Children's Experience on a Mobile EdTech Platform through a Recommender System. *Mobile Information Systems*, 2018 (1374017).

DOI: <https://doi.org/10.1155/2018/1374017>

© 2018 Almudena Ruiz-Iniesta et al.



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

## Research Article

# Improving Children's Experience on a Mobile EdTech Platform through a Recommender System

**Almudena Ruiz-Iniesta,<sup>1</sup> Luis Melgar,<sup>2</sup> Alejandro Baldominos ,<sup>3,4</sup> and David Quintana <sup>4</sup>**

<sup>1</sup>*Universidad Internacional de La Rioja, Logroño, Spain*

<sup>2</sup>*Banco Bilbao Vizcaya Argentaria (BBVA), Bilbao, Spain*

<sup>3</sup>*Smile and Learn Digital Creations, Madrid 28043, Spain*

<sup>4</sup>*Department of Computer Science, Universidad Carlos III de Madrid, Av. de la Universidad 30, Leganés 28911, Spain*

Correspondence should be addressed to David Quintana; [dquintan@inf.uc3m.es](mailto:dquintan@inf.uc3m.es)

Received 10 June 2018; Revised 5 September 2018; Accepted 30 September 2018; Published 2 December 2018

Academic Editor: Mari C. Aguayo Torres

Copyright © 2018 Almudena Ruiz-Iniesta et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Smile and Learn is an EdTech digital publisher that offers a smart library of close to 100 educational stories and gaming apps for mobile devices aimed at children aged 2 to 10 and their families. Given the complexity of navigating the content, a recommender system was developed. The system consists of two major components: one that generates content recommendations and another that provides explanations and recommendations relevant to parents and educators. The former was implemented as a hybrid recommender system that combines three kinds of recommendations. Among these, we introduce a collaborative filtering adapted to overcome specific limitations associated with younger users. The approach described in this work was tested on real users of the platform. The experimental results suggest that this recommendation model is suitable to suggest apps to children and increase their engagement in terms of usage time and number of games played.

## 1. Introduction

Smile and Learn is an EdTech digital publisher that creates and distributes educational mobile apps for children aged 2 to 10 and their families. The company offers a smart library of around 100 stories and gaming apps to provide a complete educational experience.

In addition to the content offering, the library has a smart component that gathers user information and analyses the performance across applications to obtain a detailed snapshot. Over time, the information regarding the state and the progress can be used by parents and educators both as an early warning mechanism and to keep track of child development.

The idea of using technology to enhance learning is hardly new. However, the increasing adoption of electronic devices, both in educational settings and at home, has created new opportunities to study user behavior and

performance. This has allowed the development of a wide range of applications that perform tasks like monitoring progress [1] and student fatigue [2], scaffolding learners [3], recommending open educational resources [4], modeling learners [5], or predicting future performance [6, 7], among many others. While some of these systems work in ways that are nonintrusive and transparent to the user, others provide direct feedback to the user or the educator [8].

Smile and Learn intends to progressively add that kind of functionality to the smart library, building on the metadata derived from the interaction of the increasing number of applications. Unlike other tools with a limited scope, focused on the search for progress in a limited area like programming [9] or even wider like a generic class [10], the smart library follows a more global approach. It is closer to the aim of the system described by Lopez-Fernandez et al. [11] for the

domain of engineering students, as it covers a wide range of cognitive skills together with the learning of values.

Due to the large number of alternatives on offer, the company has felt the need to develop a recommender system to help the users navigate through the library, increasing their engagement and the diversity of apps that they use. In this paper, we will thoroughly describe the functioning of the implemented recommender system. Additionally, we will evaluate the performance of the system in terms of how recommendations suggested to users impact the number of games and playing time.

Recommender systems is a very active research field [12–14]. Among the areas of application that show more potential, we can mention Education. The research effort that is being devoted to this area is very important, and so is the output in terms of academic publications and presentations made at conferences [15, 16].

The implementation and utilization of recommender systems for educational purposes is full of challenges [17]. Their evaluation is also full of difficulties [18]. From an algorithmic point of view, there are a number of approaches that differ widely in their degree of complexity and suitability, depending on the context [19]. Among the most popular ones, we can mention a few discussed in a recent benchmarking exercise for learning environments by Kopeinik et al. [20]: Most Popular [21], Collaborative Filtering [22], Content-Based [23], Usage Context-Based Similarity [24], Base Level Learning Equation with Associative Component [25], or SUSTAIN [26].

As we will discuss in detail later, the recommender system implemented in the smart library relies on a combination of demographic and use data gathered by other elements of the platform. This is consistent with other educational systems described in the literature [27]. The system consists of two major components: one that generates user-oriented content recommendations and another that provides explanations and recommendations relevant to parents and educators. It is worth noting that the former was designed to overcome specific limitations associated with younger users, such as the difficulty to provide meaningful feedback.

The rest of the document is organized as follows: Section 2 provides a general description of the system, including an introduction to the applications contained in the smart library, and the way in which the data are acquired and processed. That will be followed by Section 3 that discusses the specifics of the recommender system implemented for the case at hand. In Section 4, we report results of the experimental analysis used to evaluate the performance of the algorithm. Finally, Section 5 will be devoted to summary and conclusions.

## 2. Smart Digital Library

In this section, we introduce the Smart Digital Library. To that end, we provide a brief description of the structure, contents, and interaction with the system, followed by details regarding the way data are captured and analyzed.

**2.1. Structure and Use.** The Smart Digital Library (SDL) is a single platform of interactive games and stories that also provides access to proprietary apps designed following the principles of the theory of multiple intelligences proposed by Howard Gardner [28]. According to it, intelligence can be differentiated into specific “modalities”, rather than being dominated by a single general ability. Hence, each app is focused on one of the specific abilities described in the theory.

There are three types of apps in SDL: games, tales, and quizzes. They reinforce multiple intelligences and cognitive skills with memory, attention, coordination, and logic. They are inspired in educational content and include a range of difficulty levels adapted to children of different ages and development levels.

The SDL is multilanguage, can be used at home, on the go, or at school and provides recommendations to children, parents, and educators.

The smart library works in the following way: children access the library through an app, downloaded by their parents or educators via Google Play (Google Play is a digital distribution service operated and developed by Google, <http://play.google.com>) or the App Store (App Store is a digital distribution platform, developed and maintained by Apple Inc., for mobile apps on its iOS operating system), which works as a sole access point and storage space for all of the games and interactive stories. From this app, parents and educators can download content for the child to use, which will be saved in the device after playing.

Each child’s progress is registered, and relevant details are made available to parents and educators through the Learning Analytics Dashboard (LAD). This component, accessible from the app itself or a website, provides information on a child’s progress and playing time, as well as recommendations for improvement. Currently, SDL has more than 11 k monthly active users and more than 1.2 M total app records.

**2.2. Data Acquisition and Processing.** Child progress is monitored by the system. The platform keeps track of the areas where the user shows better performance and also those where the child faces more difficulties. Teachers and parents are able to access the data from the platform, using the LAD.

In order to extract correlations between apps played and learning, we use a feedback platform, which provides time-stamped output of app-related events (score, failures, bonus, game mode, type of app, level, etc.). The recorded fields depend on the type of app played. The information recorded by the different types of app is the following:

- (i) Tales: time spent on reading the tale, together with the level and reading mode (type of letter, reading, listening, or pictograms).
- (ii) Quiz: quiz level, time spent on completing the quiz, number of hits, failures, and total number of questions.

- (iii) Game: hits, failures, total of possible hits, time spent on completing the game, score, bonus, and level.

These data are used to enrich the child's profile. With it, the system estimates the mastery level according to multiple intelligences. This information is then employed by the recommender system to personalize a suggestion list.

The LAD shows the progress of the child and the proportion of time that the user devotes to each application. As will be explained in Section 3.2, it has been conceived to communicate the learning stages of each player effectively and includes recommendations for parents and educators. Its design follows closely numerous insights that were provided in interviews with specialists.

The recommender system plays a key role in the platform. We describe the implemented solution in the section that follows.

### 3. Recommender System

Recommender systems (RS) in this context are used to personalize online experiences. They are based on the fact that people's tastes generally follow patterns. People tend to like things that are similar to other things they liked before, and they tend to have a similar taste to other people they are close with. Among the alternatives mentioned in the introduction, the two most popular are Content-Based (CB) and Collaborative Filtering (CF) [29]. CB employs the description of the item and a profile of the user's preferences to make recommendations. Meanwhile, CF generates recommendations based on the preferences of similar users. As stated before, research on recommender systems is a very active field. Education is one of the areas where applications show great potential.

We aim at providing personalized recommendations that enable children to interact better with the SDL. The first release of SDL showed the apps grouped by worlds (Figure 1(a)). These worlds corresponded to the seven intelligences proposed in Gardner's theory; however, this user interface design had a limitation: it was difficult for children to find the applications they wanted to play. As a consequence, they always used the same apps because they already knew where they were located.

Given the mentioned use pattern, one of the goals of the RS was to design a frictionless SDL. That is, we aim at providing interactions that control the elements that inhibit people from intuitively and painlessly achieving their goals within a digital interface. For this purpose, we proposed a new use experience design with a list of seven recommended apps at the bottom of the screen (Figure 1(b)). In Section 4, we present an analysis of the benefits of this new design.

The RS uses data collected by the feedback platform in order to estimate the preferences of the child. We have also included an explanation module in our recommender system which will provide explanations of the recommendations to parents and educators through the LAD. As we will see in Section 3.2, explanations provide confidence and trust in the RS.

In the SDL, the personalization focuses on estimating the interest that a user will have in an application. For this, we combine in a hybrid recommender system [30] several types

of algorithms, each one focused on one relevant aspect of our apps and our users as it is explained below:

- (i) Trending apps: recommends two apps from the most widely played in our SDL to new users. These recommendations often comprise apps that due to their nature are very enjoyable to play regardless of the user profile.
- (ii) Surprise of the month: recommends two apps taken from the new monthly releases.
- (iii) Collaborative filtering: recommends three apps (not yet played by the user) based on the preferences of similar users.

The results from each recommender are combined in a unique list of outcomes with no further processing, where the recommendations are listed in this order: first, the trending apps are shown. These are followed by the surprise of the month and, finally, the CF recommendation. Given that ordering might have an impact, and the fact that it has not been optimized yet, our results might understate the potential benefits of the CF recommender system.

We have limited the size of the list of recommendations to seven, since displaying a larger number could be difficult in certain screens, therefore worsening the user experience. From this list, the majority of recommendations (three out of seven) are generated by the CF recommender, whose strategy we explain in further detail in the following subsection.

*3.1. Collaborative Filtering Recommendation Strategy.* The main goal of our collaborative filtering recommender is to select the apps of greatest interest to the target child according to the preferences of other SDL users. CF is the process of evaluating items using opinions of others. In the SDL, CF techniques help to identify those apps which are more useful and of more interest to each child in particular.

We have explored the use of the well-known user-based nearest-neighbor algorithm [31] in our CF. This algorithm generates predictions for users based on ratings from similar users. First, the algorithm identifies the users that, in the past, exhibited a similar behavior. Next, it analyses their ratings to identify the items that the target user should like. These similar users are the neighbors.

We have adapted the general CF process to our domain. The neighbors are those children who have played at least one application in common with the target user. Overall, the adapted process runs as follows:

- (i) Neighborhood formation. We form the neighborhood in order to find those children who are more similar to the target child by using the data collected by the feedback platform. This step requires a proximity measure to generate like-minded peers and to select the top-K neighbors.
- (ii) Rating prediction and top-K selection. For each app played by the neighbors, we generate a prediction using neighbors' ratings. Finally, the top-K apps are selected.





FIGURE 1: Screenshot of the smart library initial screen.

Next subsections describe each step of the process.

**3.1.1. Neighborhood Formation.** The first step in the CF process is the selection of the children who are more similar to the target child, to form the neighborhood. Only children who played at least one application in common with the target child are candidates to form part of the neighborhood.

The child profile includes demographic information (e.g., age, gender, and education). Each app feedback also stores information about the IP connection, device information, and duration of the session. This information will be employed to measure the proximity of each user in order to form the neighborhood.

Traditionally, CF takes into account the ratings assigned by each user to each product to generate the neighborhood and the predictions. These RS have focused on users who are able to offer explicit feedback themselves. However, in our domain, users will be children. It is worth noting that children's patterns of attention and interaction are quite different from adults [32], and they are not capable of assigning ratings to the apps they play in a meaningful way. For this reason, the metric used to build the similarity matrix in our CF recommender system is designed to rely on implicit rather than explicit feedback.

We have chosen a similarity metric defined as the weighted sum up of two relevancies: the age similarity (*ageSim*) between the children and the number of common apps (*appSim*). In this equation, the common apps account for the fact of finding users with similar preferences, but the metric also receives feedback based on the user's age, which is relevant in the context of children, where preferences can change fast during growth. In order to calculate the age similarity, we have picked a distance-based metric using a threshold. To calculate the *appSim*, we employ the app usage records. We will take into account the applications most played by the child in relation to the total applications he has played. So, the more applications two children have in common, the more similar they are. Finally, the two indicators are combined as follows:

$$\begin{aligned} \text{similarity}(\text{Ch}_1, \text{Ch}_2) &= \alpha \cdot \text{appSim}(\text{Ch}_1, \text{Ch}_2) \\ &+ (1 - \alpha) \cdot \text{ageSim}(\text{Ch}_1, \text{Ch}_2), \end{aligned} \quad (1)$$

where  $\alpha \in [0, 1]$ . This similarity measure reports a value in  $[0, 1]$  that represents how similar two children are according to their profiles. Then, the top-K children with higher similarity values with the target child are selected as neighbors. We use the implicit feedback of these users to perform our recommendation.

**3.1.2. Prediction Computation and Top-K Selection.** Finally, we have to generate predictions for the proposed apps. For each app, we analyze the records of children in the target child's neighborhood and we compute the weighted average of ratings. The value of each weight is directly related to the similarity of the target child and the corresponding neighbor, computed during the neighborhood formation. Once we have calculated the prediction for the candidate apps, we sort them according to this value. This way, the most interesting apps for the target child (the top-K apps) will be finally suggested.

**3.2. Explanations and Recommendations for Parents and Educators.** The last component of our recommender system is the explanations module, which takes into account the characteristics of each app and the potential benefits that it could provide to the child. It attaches a message to the list of recommended apps and shows all this information in the LAD (Figure 2). Our goal is to provide feedback to educators and parents on how the use of certain apps can improve learning, based on the analysis of the child's use and progress observed. When generating the explanation for a recommended app, a phrase from among a set of predefined templates is selected and filled with information about the app and the child. This information will help parents and educators to better understand how the recommended app will influence the child's educational process. These templates differ based on the intelligence developed by the app and stage of learning, in order to provide a more customized

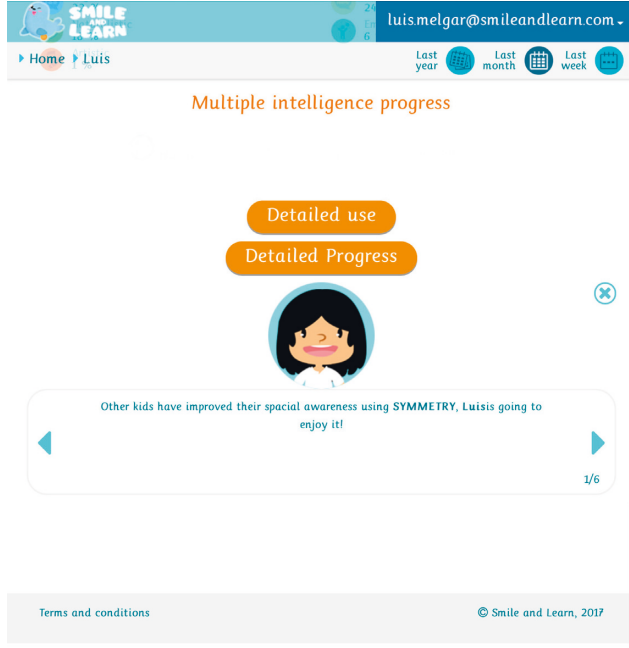


FIGURE 2: Explanations of the recommender system.

experience. A real example of this kind of explanations could be the following: “By playing *Bubbles*, Alex could develop a variety of abilities such as spatial orientation, attention, visual memory and logic-spatial reasoning.”

Displaying a plausible explanation along with a recommendation can do wonders at almost no cost, i.e., without making any changes to the underlying recommendation algorithms. The impact of explanations has been studied extensively by Tintarev and Masthoff [33]. Among the advantages, we could mention that it increases trust in the smart library, effectively helping parents and educators make good decisions in choosing the most suitable apps for their children.

#### 4. Experimental Analysis

The main purpose of the implemented recommendation system is to facilitate the access to SDL, by limiting the amount of information that the user has to go through before starting to actually play, and the making the SDL more attractive to the user by implementing personalized suggestions.

The evaluation that we present here will let us know how the patterns of usage of the SDL are affected by the recommendation system. In order to do that, we will analyze whether it has any impact on the use of the applications in the aforementioned context. Specifically, we will study the impact of the recommender system on user engagement in terms of number of games played and time spent on the system, bearing in mind that we are in a scenario in which players might have limited access to their digital devices.

The dataset used in this analysis covers the six-month period between March and August 2018, both inclusive. The SDL is used both in educational settings, like schools, and

domestic ones. Given that the interaction in the former is usually driven by teachers, and therefore, the recommender system is unlikely to have any impact, we disregarded those users. The resulting sample included 4,876 unique users and 78 applications. Finally, we filtered out the users for which there was incomplete information, leaving us with a final 4,712 users.

We only considered the applications available in the SDL that can be suitably described as interactive apps; this distinction is pertinent because the applications that were left out of the study have very different interaction dynamics (most of them can be considered tales, as the content is limited to a story).

We proceed to give a detailed description of the metrics, as well as the corresponding results. As we mentioned, we will focus on two aspects of engagement.

The formal definition of the average number of games per recommended app for user  $i$ ,  $ANG_{Ri}$ , is defined by the following expression:

$$ANG_{Ri} = \frac{Games_{Ri}}{NumApps_{Ri}}, \quad (2)$$

where  $Games_{Ri}$  is the total number of games played by user  $i$  on apps recommended to her, and  $NumApps_{Ri}$  is the total number of apps recommended to  $i$ . For this sake, we define “game” as an event where the user has completed a level and obtained a score that is registered.

The complementary metric of  $ANG_{Ri}$  would be  $ANG_{NRi}$ . The only difference between them is the fact that the latter would consider the total number of nonrecommended apps, and the number of times that they were played by the user.

The second indicator of engagement is defined in terms of time, rather than the number of games. We therefore define the average game time per recommended app for user  $i$ ,  $AGT_{Ri}$ , as

$$AGT_{Ri} = \frac{GameTime_{Ri}}{NumApps_{Ri}}, \quad (3)$$

where  $GameTime_{Ri}$  is the total accumulated time (in seconds) spent by user  $i$  playing apps recommended to her, and  $NumApps_{Ri}$  is the total number of apps recommended to  $i$ .

Once again, the complementary metric,  $GameTime_{NRi}$ , considers the total number of nonrecommended apps and the total time accumulated by user  $i$  playing them.

The experimental results are summarized in Table 1. There, we report the mean, median, and standard deviation by recommendation for the two indicators over the six-month period from the beginning of March 2018 to the end of August. As we can see, the evidence supports the positive impact of the recommendation system on engagement. There seems to be a strong positive connection between being recommended and both a higher number of games and more time spent using the SDL.

The statistical significance of the reported differences between the metrics for the recommended apps and the nonrecommended ones was formally tested. Once the null hypothesis of normality was rejected for the distribution of the four indicators using Lilliefors test ( $p < 0.001$ ), we applied the Wilcoxon test to assess the differences. These were

TABLE 1: Engagement metrics for game apps by recommendation for the period 03/01/2018–08/31/2018.

Engagement metric	Mean	Median	Std. dev.
Games			
ANG <sub>NRi</sub>	0.43	0.18	0.88
ANG <sub>Ri</sub>	2.22	1.05	5.81
Time			
AGT <sub>NRi</sub>	74.79	18.66	300.21
AGT <sub>Ri</sub>	292.84	103.49	2005.45

TABLE 2: Engagement metrics for game apps by recommendation and for the period 1/03/2018–31/08/2018. Metrics grouped by quartile of usage intensity.

Quartile	Number		Time	
	ANG <sub>NRi</sub>	ANG <sub>Ri</sub>	AGT <sub>NRi</sub>	AGT <sub>Ri</sub>
Q4	0.07	0.58	4.57	41.05
Q3	0.18	1.46	15.09	101.53
Q2	0.37	2.21	37.79	199.85
Q1	1.11	4.61	241.19	746.34

found to be significant at the 1% conventional level ( $p < 0.001$ ).

In order to provide a better understanding of the importance of the RS, we report the engagement metrics by usage intensity in Table 2.

We identified the total time spent playing games by the users of the SDL, regardless of whether the apps were recommended or not, and classified them by quartiles. As we can see, recommendations have more impact in absolute terms on the most active users. However, among the least active users, who sometimes limit their activity to opening one or two games for a little time and then proceed to uninstall the software, the effects in relative terms tends to be more important. These users, that do not know yet the library and have not had the chance to identify games that they like, often rely on the suggestions of the recommender to start the exploration of the SDL.

## 5. Summary and Conclusions

Smile and Learn’s digital library includes around 100 educational stories and gaming apps aimed at children aged 2 to 10 and their families. Given the complexity of navigating the content, an RS was introduced to ease the process-improving engagement.

Recommender systems are often based on explicit user feedback. This, however, is a factor, as the approach requires users who are sophisticated enough to provide it. This imposes a major limitation in the children’s educational technology space, for the patterns of attention and interaction with the apps are quite different from those of adults. In particular, children have a limited ability in terms of assigning ratings to the apps that they play. In order to overcome his problem, we introduce an approach that takes into account the above issues in order to enhance children’s experience with the SDL.

The RS that we present combines three kinds of recommendations. Among these, we introduced one based on collaborative filtering aimed at inferring what can be of potential interest to users considering past user-item interactions. The approach, based on implicit feedback, adapts the standard algorithm to the children’s educational app domain as a two-step process that starts with a neighborhood formation and ends with a rating prediction and top-K selection. The neighborhood formation is based on a similarity metric, tailored to this domain that considers both age and the number of apps in common among users. The RS also includes an explanations module for parents and educators.

In this context, we have studied in Section 4 to what extent has the RS achieved the goal of increasing engagement. The results support a direct connection between the use of the RS and two closely related aspects of this construct for gaming content: number of games and playing time. Recommended apps show a higher average number of games per recommended app than the nonrecommended ones, and the average game time per recommended app by user shows the same pattern. The positive impact of the RS is both substantial and statistically significant.

Even though we found different patterns depending on the intensity of use, the RS has a positive impact regardless of whether the users were among the most or the least active ones. The difference was that the impact in absolute terms was higher among the most frequent users, whereas the impact on the least frequent ones was higher in relative terms. More casual users rely more on the recommendations for their initial exploration.

This latter aspect is relevant, as it suggests that the first contact with the SDL can be managed effectively to favor positive experiences. Given that, according to the data, to an important extent, the first selections do not come from a rational analysis of the whole catalog of the SDL, but an impulsive choice based on the alternatives offered by the RS, the system could focus the selection presented to new users on trending apps. This possibility, yet to be tested, that would disregard the other two algorithms, could potentially bias the selection to increase the likelihood of a satisfying experience, hence improving engagement.

We are currently exploring new ways of personalization by aggregating information associated with the competency level of users in each of the cognitive areas. In the future, we will create a measure of profile-similarity based on the aforementioned capabilities in order to be able to provide recommendations that make the learning process more effective. This profile-similarity measure will be employed to recognize different learning styles and playing habits mining their feedback records. This approach can also be used in a clustering process that considers children habits and interests to obtain better personalized recommendations.

Finally, this longer-term vision will be complemented by near-term progress in two fronts: the optimization of the process in charge of combining and representing the output of the three subsystems of the described hybrid RS and benchmarking its performance against other state-of-the-art algorithms.



## Data Availability

The usage data used to support the findings of this study were supplied by Smile and Learn under license and so cannot be made freely available. Data are available from Alejandro Baldominos at [alejandro.baldominos@smileandlearn.com](mailto:alejandro.baldominos@smileandlearn.com) with the permission of Smile and Learn, upon reasonable request.

## Disclosure

All the authors are or were affiliated with Smile and Learn either as employees or advisors. Neither Smile and Learn nor the funding sponsors had any role in the design of the study, in the analyses or interpretation of data, in the writing of the manuscript, and in the decision to publish the results.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This research has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 756826.

## References

- [1] S. Charleer, A. V. Moere, J. Klerkx, K. Verbert, and T. D. Laet, "Learning Analytics dashboards to support adviser-student dialogue," *IEEE Transactions on Learning Technologies*, vol. 11, no. 3, pp. 389–399, 2017.
- [2] S. Gonçalves, M. Rodrigues, D. Carneiro, F. Fdez-Riverola, and P. Novais, "Boosting learning: non-intrusive monitoring of student's efficiency," *Advances in Intelligent Systems and Computing*, vol. 374, pp. 73–80, 2015, Conference of 1st International Conference on Methodologies and Intelligent Systems for Technology Enhanced Learning (MIS4TEL 2015), June 2015.
- [3] M. Ueno and Y. Miyazawa, "IRT-based adaptive hints to scaffold learning in programming," *IEEE Transactions on Learning Technologies*, p. 1, 2017.
- [4] A. Ruiz-Iniesta, G. Jiménez-Díaz, and M. Gómez-Albarrán, "A semantically enriched context-aware OER recommendation strategy and its application to a computer science OER repository," *IEEE Transactions on Education*, vol. 57, no. 4, pp. 255–260, 2014.
- [5] T. Käser, S. Klingler, A. G. Schwing, and M. Gross, "Dynamic bayesian networks for student modeling," *IEEE Transactions on Learning Technologies*, vol. 10, no. 4, pp. 450–462, 2017.
- [6] M. Al-Saleem, N. Al-Kathiry, S. Al-Osimi, and G. Badr, "Mining educational data to predict students' academic performance," *Lecture Notes in Computer Science*, vol. 9166, pp. 403–414, 2015, Conference of 11th International Conference on Machine Learning and Data Mining in Pattern Recognition (MLDM 2015), July 2015.
- [7] H. Bydžovská, "Student performance prediction using collaborative filtering methods," *Lecture Notes in Computer Science*, vol. 9112, pp. 550–553, 2015, Conference of 17th International Conference on Artificial Intelligence in Education (AIED 2015), June 2015.
- [8] R. Bodily and K. Verbert, "Review of research on student-facing learning Analytics dashboards and educational recommender systems," *IEEE Transactions on Learning Technologies*, vol. 10, no. 4, pp. 405–418, 2017.
- [9] M. L. Barrón-Estrada, R. Zatarain-Cabada, F. G. Hernández, R. O. Bustillos, and C. A. Reyes-García, "An affective and cognitive tutoring system for learning programming," in *Advances in Artificial Intelligence and Its Applications: 14th Mexican International Conference on Artificial Intelligence, MICAI 2015*, O. Pichardo Lagunas, O. Herrera Alcántara, and G. Arroyo Figueroa, Eds., vol. 9414, pp. 171–182, Lecture notes in Computer Science, Springer International Publishing, Cuernavaca, Morelos, Mexico, October 2015.
- [10] M. H. Dlab and N. Hoić-Božić, "Increasing Students' Academic Results in e-course Using Educational Recommendation Strategy," in *Proceedings of the 17th International Conference on Computer Systems and Technologies 2016 (CompSysTech '16)*, A. Smrikarov and B. Rachev, Ed., pp. 391–398, ACM, New York, NY, USA, 2016.
- [11] D. Lopez-Fernandez, P. Alarcon, and E. Tovar, "Assessment and development of transversal competences based on student's autonomous learning," in *Proceedings of 2016 IEEE Global Engineering Education Conference (EDUCON)*, pp. 482–487, IEEE Computer Society, Abu Dhabi, UAE, April 2016.
- [12] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, 2005.
- [13] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowledge-Based Systems*, vol. 46, pp. 109–132, 2013.
- [14] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, "Recommender system application developments: a survey," *Decision Support Systems*, vol. 74, pp. 12–32, 2015.
- [15] H. Drachsler, K. Verbert, O. C. Santos, and N. Manouselis, "Panorama of recommender systems to support learning," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, and B. Shapira, Eds., pp. 421–451, Springer US, Boston, MA, USA, 2015.
- [16] M. Erdt, A. Fernández, and C. Rensing, "Evaluating recommender systems for technology enhanced learning: a quantitative survey," *IEEE Transactions on Learning Technologies*, vol. 8, no. 4, pp. 326–344, 2015.
- [17] J. Tarus and Z. Niu, "A survey of learner and researcher related challenges in e-learning recommender systems," *Communications in Computer and Information Science*, vol. 734, pp. 122–132, 2017, Conference of 6th International Workshop on Learning Technology for Education Challenges (LTEC 2017), August 2017.
- [18] M. Lombardi and A. Marani, "A comparative framework to evaluate recommender systems in technology enhanced learning: a case study," *Lecture Notes in Computer Science*, vol. 9414, pp. 155–170, 2015, Conference of 14th Mexican International Conference on Artificial Intelligence (MICAI 2015), October 2015.
- [19] I. Portugal, P. Alencar, and D. Cowan, "The use of machine learning algorithms in recommender systems: a systematic review," *Expert Systems with Applications*, vol. 97, pp. 205–227, 2018.
- [20] S. Kopeinik, D. Kowald, and E. Lex, "Which algorithms suit which learning environments?," in *Proceedings of A Comparative Study of Recommender Systems in TEL. Adaptive and Adaptable Learning: 11th European Conference on Technology Enhanced Learning (EC-TEL 2016)*, K. Verbert, M. Sharples, and T. Klobučar, Eds., vol. 9891, pp. 124–138,



- Springer, Lecture Notes in Computer Science, Lyon, France, September 2016, 2016.
- [21] R. Jäschke, L. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme, "Tag recommendations in folksonomies," in *Proceedings of Knowledge Discovery in Databases: PKDD 2007: 11th European Conference on Principles and Practice of Knowledge Discovery in Databases*, J. N. Kok, J. Koronacki, R. Lopez de Mantaras, S. Matwin, D. Mladenić, A. Skowron, Eds., vol. 4702, pp. 506–514, Lecture Notes in Computer Science, Springer Berlin Heidelberg, Warsaw, Poland, September 2007.
  - [22] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web: Methods and Strategies of Web Personalization. Lecture Notes in Computer Science*, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds., vol. 4321 Springer Berlin Heidelberg, Berlin, Heidelberg, , pp. 291–324, 2007.
  - [23] J. Basilico and T. Hofmann, "Unifying collaborative and content-based filtering," in *Proceedings of Twenty-first International Conference on Machine Learning ICML '04*, p. 9, ACM, Banff, Alberta, Canada, July 2004.
  - [24] K. Niemann and M. Wolpers, "Usage context-boosted filtering for recommender systems in TEL," in *Proceedings of Scaling up Learning for Sustained Impact: 8th European Conference, on Technology Enhanced Learning (EC-TEL 2013)*, D. Hernández-Leo, T. Ley, R. Klamma, and A. Harre Eds., vol. 8095, pp. 246–259, Lecture Notes in Computer Science, Springer Berlin Heidelberg, Paphos, Cyprus, September 2013.
  - [25] D. Kowald, S. Kopeinik, P. Seitlinger, T. Ley, D. Albert, and C. Trattner, "Refining frequency-based tag reuse predictions by means of time and semantic context," in *Proceedings of Mining, Modeling, and Recommending 'Things' in Social Media: 4th International Workshops, MUSE 2013 and MSM 2013*, M. Atzmueller, A. Chin, C. Scholz, and C. Trattner, Eds., Lecture Notes in Computer Science, Springer International Publishing, Cham, Vol. 8940, pp. 55–74, Prague, Czech Republic and Paris, France, September 2013 and May 2013.
  - [26] B. Love, D. Medin, and T. Gureckis, "Sustain: a network model of category learning," *Psychological Review*, vol. 111, no. 2, pp. 309–332, 2004.
  - [27] M. El Mabrouk, S. Gaou, and M. Rtili, "Towards an intelligent hybrid recommendation system for e-learning platforms using data mining," *International Journal of Emerging Technologies in Learning*, vol. 12, no. 6, pp. 52–76, 2017.
  - [28] H. E. Gardner, *Intelligence reframed: Multiple Intelligences for the 21st Century*, Basic Books, Hachette, UK, 2000.
  - [29] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, *Recommender Systems Handbook*, Springer, Berlin, Germany, 2015.
  - [30] R. Burke, "Hybrid recommender systems: survey and experiments," *User modeling and user-adapted interaction*, vol. 12, pp. 331–370, 2002.
  - [31] P. Brusilovski, A. Kobsa, and W. Nejdl, *The Adaptive Web: Methods and Strategies of Web Personalization*, Springer Science & Business Media, Vol. 4321, Springer Science & Business Media, Berlin, Germany, 2007.
  - [32] Y. Deldjoo, C. Frà, M. Valla et al., "Enhancing Children's experience with recommendation systems," in *Proceedings of 11th ACM Conference of Workshop on Children and Recommender Systems (KidRec'17)*, Milan, Italy, August 2017.
  - [33] N. Tintarev and J. Masthoff, "A survey of explanations in recommender systems," in *Proceedings of 2007 IEEE 23rd International Conference on Data Engineering Workshop*, pp. 801–810, Istanbul, Turkey, April 2007.